CLAIMS

I claim:

1. An electronic voting system for use with a computerized network, comprising:

a plurality of voting computers coupled to the computerized network, wherein each voting computer provides an electronic encrypted ballot, wherein each electronic ballot is encrypted under a discrete log asymmetric encryption process using underlying groups Z_p or elliptic curve;

at least first, second and third authority computers coupled to the computerized network, wherein the first authority computer is configured to receive a series of electronic ballots corresponding to an aggregation of each of the electronic ballots received from the plurality of voting computers, and to apply a secret, one-way cryptographic transformation using at least a first secret key to anonymously shuffle the series of electronic ballots and produce a first shuffled series of ballots, wherein only the first authority computer knows a correspondence between the first series of shuffled ballots and the series of electronic ballots, and wherein the first authority computer is further configured to provide a first linear size, non-interactive proof of correctness for the first series of shuffled ballots based on a scaled iterated logarithmic multiplication proof;

wherein the second authority computer is configured to receive the first series of shuffled ballots, to apply the cryptographic transformation using at least a second secret key to anonymously shuffle the first series of shuffled ballots and produce a second series of shuffled ballots, wherein only the second authority computer knows a correspondence between the first series of shuffled ballots and the second series of shuffled ballots, and wherein the second authority computer is

further configured to provide a second linear size, non-interactive proof of correctness for the second series of shuffled ballots based on the scaled iterated logarithmic multiplication proof;

wherein the third authority computer is configured to receive the second series of shuffled ballots, to apply the cryptographic transformation using at least a third secret key to anonymously shuffle the second series of shuffled ballots and produce a third series of shuffled ballots, wherein only the third authority computer knows a correspondence between the third series of shuffled ballots and the second series of shuffled ballots, and wherein the third authority computer is further configured to provide a third linear size, non-interactive proof of correctness for the third series of shuffled ballots based on the scaled iterated logarithmic multiplication proof; and

a verification computer coupled to the computerized network, wherein the verification computer is configured to receive the proofs of correctness from the first, second and third authority computers and without interacting with the first, second and third authority computers, to verify a correctness of the shuffled ballots.

2. The system of claim 1, further comprising:

a server computer system coupled to the computerized network, wherein the server computer system is configured to: receive the plurality of electronic ballots from the plurality of voting computers; verify a proof of validity of each of the plurality of electronic ballots; form an encrypted tally of the votes from the plurality of electronic ballots; transmit the encrypted tally to the first, second and third authority computers; receive ballot decryption shares produced from at least two of the first, second and third authority computers; and compute a decrypted tally; and

at least one voting poll computer coupled to the computerized network and providing some of the plurality of electronic encrypted ballots to the server computer system.

3. The system of claim 1 wherein the first, second and third authority computers are configured to provide Chaum-Pedersen proofs for the first, second and third shuffles of the ballots, respectively, and wherein each of the first, second and third authority computers generate an initial challenge series, receive a challenge from at least one verification computer, and generate the cryptographic transformation based on an exponentiation of the initial and received challenges.

4. The system of claim 1 wherein the computerized network includes the World Wide Web, wherein each of the plurality of voting computers and first, second and third authority computers include a web browser program.

5. The system of claim 1 wherein the plurality of voter computers include at least one palm-sized computer, cell phone, wearable computer, interactive television terminal or Internet appliance.

6. A computer system for receiving a sequence of elements, comprising:

a server computer coupled to a computer network and configured to:

receive a sequence of electronic data elements representing individual data files,

apply a cryptographic transformation using at least a first secret key to anonymously permute the sequence of electronic data elements and produce a first shuffled sequence of electronic

data elements, wherein the server computer knows a correspondence between the first shuffled sequence of electronic data elements and the sequence of electronic data elements, and

generate a first linear size proof of correctness for the first shuffled sequence of electronic data elements based on a scaled iterated logarithmic multiplication proof.

7. The system of claim 6 wherein the received sequence of electronic data elements are encrypted using Z_p or elliptic curve groups using a key unknown to the server computer, and wherein the server computer is further configured to:

receive a series of randomly generated values e_i from a verifier computer;

secretly generate a series of values U_i based on a secret, one-way cryptographic transformation that employs the received series of values e_i and secretly generated values

$$\bar{u}_i$$

permute the sequence of electronic data elements to produce the first shuffled sequence of elements based on the series of values U_i and a secret value d; and

provide the values U_i and a series of proof values based on the cryptographic transformation as a proof of knowledge that the server computer has access to how the cryptographic transformation permuted the sequence of electronic data elements to produce the first shuffled sequence of elements without revealing the cryptographic transformation to the verifier computer.

8. The system of claim 6 wherein the server computer is further configured for:

receiving a plurality of public keys from a corresponding plurality of individuals, wherein each of the plurality of individuals have a private key corresponding to one of the plurality of public keys;

receiving a request for a certificate from one of the plurality of individuals having a one private key;

providing at least a subset of the plurality of public keys to the requesting individual;

receiving a shuffle of the plurality of public keys and a linear size proof of correctness for the shuffled public keys based on a scaled iterated logarithmic multiplication proof and a value corresponding to the one private key, wherein the value provides proof that the one individual has knowledge of the one private key without revealing the one private key;

checking the proof of correctness;

checking that the value is mathematically related to a one of the public keys that corresponds to the one private key;

issuing a certificate to the one individual; and

reducing the plurality of public keys by the one public key.

9. The system of claim 6 wherein the sequence of electronic elements are public keys, and wherein the server if further configured to check, in response to a request from an individual, that the individual has a value uniquely and mathematically related to a one of the public keys; and

if so, issue a certificate to the one individual.

10. A computer-implemented method, comprising:

receiving a plurality of public keys from a corresponding plurality of individuals, wherein each of the plurality of individuals have a private key corresponding to one of the plurality of public keys;

receiving a request for a certificate from one of the plurality of individuals having a one private key;

providing at least a subset of the plurality of public keys to the requesting individual;

receiving a shuffle of the plurality of public keys and a linear size proof of correctness for the shuffled public keys based on a scaled iterated logarithmic multiplication proof and a value corresponding to the one private key, wherein the value provides proof that the one individual has knowledge of the one private key without revealing the one private key;

checking the proof of correctness;

checking that the value is mathematically related to a one of the public keys that corresponds to the one private key;

issuing a certificate to the one individual; and

reducing the plurality of public keys by the one public key.

11. The method of claim 10 wherein the method further includes setting a value G to a subgroup operator g from an Z_p or elliptic curve group, wherein providing at least a subset of the plurality of public keys includes providing all of the then current public keys H.

12. The method of claim 10 wherein providing at least a subset of the plurality of public keys includes providing at least a subset of a plurality of public key pairs, wherein receiving a shuffle of the plurality of public keys includes receiving a shuffle of a true subset of the plurality of public key pairs as selected by the one individual.

13. The method of claim 10, further comprising:

receiving from each of a plurality of authorities, in sequence, a shuffled set of the plurality of public keys H' based on a secret cryptographic shuffle operation performed on at least a subset of the plurality of public keys to produce the shuffled set of the plurality of public keys H';

receiving from each of a plurality of authorities, in sequence, a verification transcript of the cryptographic shuffle operation; and

verifying a correctness of the cryptographic shuffle operation based on the verification transcript; and if verified, then setting at least a subset of the plurality of public keys to H to H'.

14. The method of claim 10 , further comprising:

at a time after receiving at least some of the plurality of public keys, setting at least a subset of the then received plurality of public keys to a received shuffled set of the plurality of public keys, wherein the shuffled set of the plurality of public keys have been received from a third party.

15. The method of claim 10, further comprising:

receiving the issued certificate from the one of the plurality of individuals; and

providing an electronic ballot to the one individual.

16. The method of claim 10 wherein issuing a certificate includes digitally signing the received request to produce a public key infrastructure ("PKI") certificate.

17. The method of claim 10 , further comprising:

receiving issued certificates from at least some of the plurality of individuals and providing initial electronic ballots in response thereto; and

receiving unencrypted voted ballots from the at least some of the plurality of individuals.

18. A computer-implemented cryptographic method between a prover computer and a verifier computer, the method comprising:

selecting a subgroup generator g selected from a group G;

secretly generating a prover key c, and a commitment value C based on the subgroup generator g;

secretly establishing a cryptographic relationship between first and second sequences of elements;

providing to the verifier computer the commitment C and the first and second sequences of elements, but not the cryptographic relationship;

computing a series of proof values based on the cryptographic relationship; and

providing the series of computed proof values to the verifier computer as a non-interactive proof of knowledge that the prover computer has access to the cryptographic relationship without revealing the cryptographic relationship to the verifier computer.

19. The method of claim 18 wherein at least the second sequence of elements is a sequence of encrypted ballots, wherein each ballot is encrypted using Z_p or elliptic curve groups;

wherein the first and second sequences of elements are respectively

$$(X_1,\ldots,X_k) \quad \textit{and} \quad (Y_1,\ldots,Y_k)$$

wherein the first and second sequence of elements have the cryptographic relationship

and wherein computing and providing the series of proof

$$(g^{u_1},\ldots,g^{u_k}) = (X_1,\ldots,X_k)$$
$$(g^{v_1},\ldots,g^{v_k}) = (Y_1,\ldots,Y_k) \text{ and where}$$

$$c^k \prod_{i=1}^{k} u_i = \prod_{i=1}^{k} v_i$$

values includes providing Chaum-Pedersen proofs based on:

for each $0 \le i \le k$ generate random r_i

$$R_i = g^{r_i}$$

for each $1 \le i \le k$ $w_i = r_i u_i / r_{i-1}$

$$W_i = g^{w_i}$$
$$z_i = w_i / v_i$$
$$Z_i = g^{z_i}$$

wherein the Chaum-Pedersen proofs provided to the verifier computer are of a form:

$$(R_{i-1}, X_i, R_i, W_i) \text{ and } (Y_i, C, W_i, Z_i).$$

20. The method of claim 18, further comprising:

permuting the first sequence of elements to produce the second sequence of elements based on a cryptographic transformation;

receiving a randomly generated value t from the verifier computer;

secretly generating a value T based on the received value t and the subgroup generator, and secretly generating a value S based on the received value t and the prover key c; and

wherein computing and providing to the verifier computer the series of proof values includes providing a series of values based on the cryptographic transformation as a proof of knowledge that the prover computer has access to how the cryptographic transformation permuted the first sequence of elements to produce the second sequence of elements without revealing the cryptographic transformation to the verifier computer.

21. The method of claim 18, further comprising:

permuting the first sequence of elements to produce the second sequence of elements based on a cryptographic transformation in a form of

$$(g^{u_1},\ldots,g^{u_k}) = (X_1,\ldots,X_k)$$
$$(g^{cu_{\pi(1)}},\ldots,g^{cu_{\pi(k)}}) = (Y_1,\ldots,Y_k)$$

receiving a randomly generated value t from the verifier computer;

secretly generating a value T based on raising the subgroup generator g to the received value t, and secretly generating a value S based on raising the value T to the prover key c; and

wherein computing and providing to the verifier computer the series of proof values includes providing a series of values based on the cryptographic transformation in a form of:

$$U_i = X_i / T$$
$$V_i = Y_i / S$$

as a proof of knowledge that the prover computer has access to how the cryptographic transformation permuted the first sequence of element to provide the second sequence of elements without revealing the cryptographic transformation to the verifier computer.

22. The method of claim 18, further comprising:

receiving the first sequence of elements as a set of elements that have previously been permuted in a manner unknown to the prover computer;

receiving a series of randomly generated values e_i from the verifier computer;

secretly generating a series of values U_i based on a secret cryptographic transformation that employs the received series of values e_i and secretly generated values

$\bar{u}_i$

permuting the second sequence of elements with respect to the first sequence of elements based on the series of values U_i and a secret value d; and

wherein computing and providing to the verifier computer the series of proof values includes providing the resulting values U_i and providing a series of proof values based on the cryptographic transformation as a proof of knowledge that the prover computer has access to how the cryptographic transformation permuted the first sequence of element to provide the second sequence of elements without revealing the cryptographic transformation to the verifier computer.

23. The method of claim 18, further comprising:

receiving the first sequence of elements as a set of elements that have previously been permuted in a manner unknown to the prover computer;

receiving a series of randomly generated values e_i from the verifier computer;

secretly generating a series of values U_i based on a secret cryptographic transformation of a form

$$u_i = \bar{u}_i + e_i = \log_g U_i$$

permuting the second sequence of elements with respect to the first sequence of elements based on the series of values U_i and a secret value d based on the following operations

$$(V_1,\ldots,V_k) = (U^d_{\pi(1)},\ldots,U^d_{\pi(k)})$$
$$D = g^d$$
$$v_i = \log_g V_i$$
$$A_i = X_i^{v_i}$$
$$B_i = Y_i^{u_i}$$

and wherein computing and providing to the verifier the series of proof values includes providing the resulting values U_i,

$$A = \prod_{i=1}^{k} A_i$$
$$B = \prod_{i=1}^{k} B_i$$

and for $1 \leq i \geq k$, providing a series of proof Chaum-Pedersen of a form

$$(g, V_i, X_i, A_i) \quad \text{and} \quad (g, U_i, Y_i, B_i)$$

and a Chaum-Pedersen proof for (D, A, C, B) as a proof of knowledge that the prover computer has access to how the cryptographic transformation permuted the first sequence of element to provide the second sequence of elements without revealing the cryptographic transformation to the verifier computer.

24. The method of claim 23, further comprising repeating the receiving the first sequence of elements, receiving a series of randomly generated values, secretly generating a series of values, and permuting the

second sequence of elements for l-tuple of elements in the first sequence of elements.

25. The method of claim 22 wherein receiving the first sequence of elements includes receiving a subset of a set of identifying elements, wherein each identifying element in the set corresponds to an individual, and wherein the method further comprises:

receiving an anonymous certificate if the verifying computer verifies the series of proofs.

26. The method of claim 18 wherein the group G is Z_p.

27. The method of claim 18 wherein the group G is an elliptic curve group.

28. A computer-readable medium whose contents provide instructions, when implemented by a computer, perform a shuffling of a sequence of electronic data elements, comprising:

receive the sequence of electronic data elements;

apply a secret, one-way cryptographic transformation using at least a first secret key to anonymously permute the sequence of electronic data elements and produce a first shuffled sequence of electronic data elements; and

generate a first linear size, non-interactive proof of correctness for the first shuffled sequence of electronic data elements based on a scaled iterated logarithmic multiplication proof.

29. The computer-readable medium of claim 28 wherein the received sequence of electronic data elements are encrypted with an underlying mathematical group being a ring of integers having a modulus integer value p (Z_p).

30. The computer-readable medium of claim 28 wherein the computer-readable medium is a logical node in a computer network receiving the sequence of electronic data elements and the contents.

31. The computer-readable medium of claim 28 wherein the computer-readable medium is a computer-readable disk.

32. The computer-readable medium of claim 28 wherein the computer-readable medium is a data transmission medium transmitting a generated data signal containing the contents.

33. The computer-readable medium of claim 28 wherein the computer-readable medium is a memory of a computer system.

34. The computer-readable medium of claim 28 wherein the computer-readable medium is an Internet connection link to a voting authority server computer.

35. In a cryptographic method, a transmitted signal for use by a computer, comprising:

a shuffled sequence of electronic data elements representing individual data files, wherein a one-way cryptographic transformation using at least a first secret key anonymously permuted an input sequence of electronic data elements to produce the shuffled sequence of electronic data elements, and

a linear size proof of correctness for the shuffled sequence of electronic data elements based on a scaled iterated logarithmic multiplication proof.